

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
12 September 2002 (12.09.2002)

PCT

(10) International Publication Number
WO 02/071677 A2

(51) International Patent Classification⁷: H04L
(21) International Application Number: PCT/US01/48342
(22) International Filing Date:
16 November 2001 (16.11.2001)
(25) Filing Language: English
(26) Publication Language: English
(30) Priority Data:
09/717,877 21 November 2000 (21.11.2000) US

(71) Applicant: GLAXO GROUP LIMITED [US/US];
Glaxo Wellcome House, Berkeley Avenue, Greenford,
Middlesex UB6 0NN (GB).

(72) Inventors: ALDAM, Gary, Victor; Glaxo Wellcome
House, Berkeley Avenue, Greenford, Middlesex UB6
0NN (GB). ANDERSON, Gregory, Cox; Five Moor
Drive, Research Triangle Park, NC 27709 (US). BROWN,
Jacqueline, T.; Glaxo Wellcome House, Berkeley Ave-
nue, Greenford, Middlesex UB6 0NN (GB). CLEARY,
Elena, W.; Five Moore Drive, Research Triangle Park,
NC 27709 (US). COBB, Paul, H.; 14 Hebing End,
Benington, Hertfordshire SG2 7DD (GB). HOUSTON,
Thomas, E.; Glaxo Wellcome House, Berkeley Avenue,
Greenford, Middlesex UB6 0NN (GB). JONES, Ed-
ward, A.; Five Moore Drive, Research Triangle Park,
NC 27709 (US). LEVER, Roger, N.; Glaxo Wellcome
House, Berkeley Avenue, Greenford, Middlesex UB6
0NN (GB). LOUV, William, C.; Five Moore Drive,
Research Triangle Park, NC 27709 (US). LOWMAN,
Douglas, S.; Five Moore Drive, Research Triangle Park,
NC 27709 (US). MARR, Andrew, P.; Glaxo Wellcome
House, Berkeley Avenue, Greenford, Middlesex UB6 0NN
(GB). MARTIN-HARKER, Denise, E.; Glaxo Wellcome
House, Berkeley Avenue, Greenford, Middlesex UB6 0NN
(GB). MCKEE, Karen, L.; Five Moore Drive, Reserach
Triangle Park, NC 27709 (US). MOORE, Gillian; Glaxo

Wellcome House, Berkeley Avenue, Greenford, Middlesex
UB6 0NN (GB). PARKINSON, Christopher; Glaxo
Wellcome House, Berkeley Avenue, Greenford, Middlesex
UB6 0NN (GB). PHELPS, Michele; Glaxo Wellcome
House, Berkeley Avenue, Greenford, Middlesex UB6
0NN (GB). PRITCHARD, James, F.; 3504 Proprietor
Way, Raleigh, NC 27612 (US). TANNER, Richard, J.,
N.; Glaxo Wellcome House, Berkeley Avenue, Greenford,
Middlesex UB6 0NN (GB). VAN DEN HONERT, Debo-
rah, A.; 15061 E. Aberdeen Ave., Aurora, CO 80016 (US).
WOOD, Ian, T.; 891-995 Greenford Road, Greenford,
Middlesex UB6 0HE (GB).

(74) Agents: KANAGY, James, M. et al.; SmithKline
Beecham Corporation, Corporate Intellectual Property,
UW2220, 709 Swedeland Road, P.O. Box 1539, King of
Prussia, PA 19406-0939 (US).

(81) Designated States (*national*): AE, AG, AL, AM, AT, AU,
AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU,
CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH,
GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC,
LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW,
MX, MZ, NO, NZ, PH, PL, PT, RO, RU, SD, SE, SG, SI,
SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA,
ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM,
KR, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW),
Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),
European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR,
GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent
(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR,
NE, SN, TD, TG).

Published:

— without international search report and to be republished
upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guid-
ance Notes on Codes and Abbreviations" appearing at the begin-
ning of each regular issue of the PCT Gazette.

(54) Title: INFORMATION CRBATION AND MANAGEMENT SYSTEM

(57) Abstract: A method, system and computer program product for managing and tracking information. The managed information can relate to, but is not limited to, information about products, their marketing and their sales. Information about chemicals (or groups thereof) can be tracked during the drug discovery, approval, and maintenance process. Such a system reduces the amount of data entry required compared to systems that require re-entry of data from an earlier phase to perform tasks in a later phase. The system further provides (1) a standardized way of recording information, (2) a secure and accessible environment for storing and managing the information, and (3) a system for compiling the information into a variety of regulatory dossiers and other documents by re-using the information. By re-using components (e.g., modules), administrative costs and efforts can be reduced since previously approved sub-documents can be used as a starting point for the creation of new documents. The previously approved sub-documents or modules can be managed according to a life cycle. Use of the life cycle reduces the chances that a document with errors will be released.

WO 02/071677 A2

INFORMATION CREATION AND MANAGEMENT SYSTEM

BACKGROUND OF THE INVENTION

Field of the Invention

5 The present invention is directed to a computer system, method and computer
program product for creating and managing information, and more particularly for
creating and managing information about documents in a pharmaceutical environment
in which all aspects of the drug discovery and development process are managed (e.g.,
pre-discovery, discovery, development, initial (domestic or international) regulatory
10 submissions, and (domestic or international) regulatory maintenance submissions).

Discussion of the Background

 A voluminous amount of regulatory information needs to be generated by
pharmaceutical companies in order to support international drug registrations. That
15 information generation is often divorced, however, from the analysis and re-use of that
same information and is not readily accessible to the many scientists who interact to
bring about the new drug product. Accordingly, the coordination of information on
existing and future products is often insufficient to capture the potential advantages that
a company could achieve by disseminating that information in a universal manner and
20 in a timely fashion.

 A number of document control systems exist to maintain one or more document
types. Examples of such systems include the Drug DocUMENTation System by
Michael Umen and Co. Mr. Umen has been granted two U.S. patents: (1) U.S. Patent
No. 5,963,967 entitled "Drug document production system" and U.S. Patent No.
25 5,734,883 entitled "Drug document production system." Other document systems
include Documentum by Documentum, Inc., and CoreDossier. Information about the
Documentum system can be found on its Web site.

SUMMARY OF THE INVENTION

It is an object of the present invention to provide an information creation and management system that is capable of (1) producing information about products, their marketing and their sales and (2) tracking such information. One use of the present invention is in creating and tracking information about chemicals (or groups thereof) during various phases/processes (e.g., (1) during the drug discovery, approval, and maintenance processes and (2) during the creation or modification of dossiers, plans, reports such as (2a) submissions to pricing and reimbursement authorities, (2b) submissions to U.S. state formularies, (2c) sales training material, (2d) environmental permit applications for manufacturing sites). Such a system reduces the amount of data entry required compared to systems that require re-entry of data from an earlier phase to perform tasks in a later phase.

It is another object of the present invention to provide (1) a standardized way of recording information, (2) a secure and accessible environment for storing and managing the information, and (3) a system for compiling the information into a variety of regulatory dossiers and other documents by re-using the information. The secure environment includes traceability of where re-useable components have been used and by whom changes were made (independent of the status of components/modules).

By re-using document components within the system, administrative costs and efforts can be reduced as well since previously approved sub-documents can be used as a starting point for the creation of new documents. The previously approved sub-documents or modules can be managed according to a "life cycle" (i.e., the set of rules that define how a sub-document undergoes changes and what corresponding approvals are required to authorize those changes). Use of the life cycle reduces the chances that a document with errors will be released.

It is another object of the present invention to more efficiently share information to provide improved input and feedback during a drug development by determining the non-approval related goals that must be addressed simultaneously with the approval related goals in order to properly market a drug. By including comparative product analysis statements/marketing as part of the goal of a product development effort, a

product may be more effectively marketed at approval or release time (rather than having to generate new tests after a product's approval in order to support a claim that a marketing department would like to make).

5

BRIEF DESCRIPTION OF THE DRAWINGS

A more complete appreciation of the invention and many of the attendant advantages thereof will become readily apparent with reference to the following detailed description, particularly when considered in conjunction with the accompanying drawings, in which:

10

Figure 1 is a schematic illustration of a computer for managing documents according to the present invention;

Figure 2 is a block diagram of the inter-relationships between various components of a software embodiment of the present invention;

15

Figure 3 is a captured user interface (i.e., a screenshot of a user interface) for displaying and/or controlling information;

Figure 4 is a screenshot illustrating an exemplary series of database entries that are combined to dynamically provide a status of an information plan;

20

Figure 5 is a screenshot illustrating an exemplary series of database entries that are combined to dynamically provide a list of needs, evidence, and activities of an information plan;

Figure 6 is a block diagram of the components of two logical sub-levels for Level 2 of Figure 2;

Figure 7 is a block diagram of the relative job functions performed by various groups/people in the authoring process;

25

Figure 8A is a screenshot showing a customization interface for a template repository;

Figure 8B is a screenshot showing how the customization interface of Figure 8A can update a user's preferences;

30

Figure 9 is a screenshot illustrating template management and usage functions according to one aspect of the present invention;

Figure 10A is a screenshot illustrating an exemplary user interface for retrieving stored templates and their corresponding guides;

Figures 10B and 10C are legends describing status and file types of the templates illustrated in Figure 10A;

5 Figure 10D is an exemplary interface for displaying the details of a template by selecting the "Details" icon of Figure 10A;

Figure 10E is an exemplary interface for displaying the details of guidance;

Figure 11 is a screenshot illustrating a search interface according to one aspect of the present invention;

10 Figure 12 is a screenshot illustrating a search result obtained using the search interface of Figure 11;

Figure 13 is a screenshot illustrating a filter interface according to one aspect of the present invention;

15 Figure 14 is a screenshot illustrating a site feedback interface according to one aspect of the present invention;

Figure 15 is a screenshot illustrating a help interface according to one aspect of the present invention;

Figure 16A is a screenshot illustrating a management interface of an administration tool according to one aspect of the present invention;

20 Figure 16B is a legend explaining the icons of Figure 16A;

Figure 17 is a screenshot illustrating the management interface of Figure 16A being used to add a new template;

Figure 18 is a screenshot illustrating the management interface of Figure 16A being used to search for an existing template;

25 Figure 19 is a screenshot illustrating the management interface of Figure 16A being used to select how to modify an existing template;

Figure 20 is a screenshot illustrating the management interface of Figure 16A being used to modify an existing template;

30 Figure 21 is a screenshot illustrating the management interface of Figure 16A being used to delete an existing template;

Figure 22 is a screenshot illustrating a report interface for use in conjunction with the management interface of Figure 16A;

Figure 23A is a series of needs to be met according to an exemplary project or report; and

5 Figure 23B is a series of outputs to be tracked in order to fulfill the project or report of Figure 23A.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring now to the drawings, in which like reference numerals designate
10 identical or corresponding parts throughout the several views, Figure 1 is a schematic illustration of a computer system for creating and managing information (e.g., information corresponding to an information plan (i.e., the definitions of the goals to be achieved, the evidence needed to support that the goals have been met, and the issues involved in meeting those goals) (e.g., for use in creating a document relating to drug
15 discovery and/or a regulatory submission)). A computer 100 implements the method of the present invention, wherein the computer housing 102 houses a motherboard 104 which contains a CPU 106, memory 108 (e.g., DRAM, ROM, EPROM, EEPROM, SRAM, SDRAM, and Flash RAM), and other optional special purpose logic devices (e.g., ASICs) or configurable logic devices (e.g., GAL and reprogrammable FPGA).
20 The computer 100 also includes plural input devices, (e.g., a keyboard 122 and mouse 124), and a display card 110 for controlling monitor 120. In addition, the computer system 100 further includes a floppy disk drive 114; other removable media devices (e.g., compact disc 119, tape, and removable magneto-optical media (not shown)); and a hard disk 112, or other fixed, high density media drives, connected using an
25 appropriate device bus (e.g., a SCSI bus, an Enhanced IDE bus, or a Ultra DMA bus). Also connected to the same device bus or another device bus, the computer 100 may additionally include a compact disc reader 118, a compact disc reader/writer unit (not shown) or a compact disc jukebox (not shown). Although compact disc 119 is shown in a CD caddy, the compact disc 119 can be inserted directly into CD-ROM drives that
30 do not require caddies. In addition, a printer (not shown) also provides printed copies of reports (e.g., drug discovery or regulatory reports).

As stated above, the system includes at least one computer readable medium. Examples of computer readable media are compact discs 119, hard disks 112, floppy disks, tape, magneto-optical disks, PROMs (EPROM, EEPROM, Flash EPROM), DRAM, SRAM, SDRAM, etc. Stored on any one or on a combination of computer readable media, the present invention includes software for controlling both the hardware of the computer 100 and for enabling the computer 100 to interact with a human user. Such software may include, but is not limited to, device drivers, operating systems and user applications, such as development tools. Such computer readable media further includes the computer program product of the present invention for generating and tracking information. The computer code devices of the present invention can be any interpreted or executable code mechanism, including but not limited to scripts (including Active Server pages), interpreters, dynamic link libraries, Java classes, and complete executable programs. These computer code devices may run on either one of, or on a combination of, a client and a server computer. Information on providing Web services is provided in the following references which are incorporated herein by reference: (1) Visual Studio Core Reference Set, by Microsoft Press, (2) Visual InterDev 6.0: Web Technologies Reference, by Microsoft Press, (3) Professional Active Server Pages 2.0 by Francis et al., published by WROX Press Ltd., (4) Oracle PL/SQL Programming by Scott Urman, Published: March 1996, (5) Hitchhikers Guide to Visual Basic and SQL Server: with CD-ROM, by William Vaughn, Published: May 1997, (6) Using Microsoft SQL Server 6.5 (Special Edition) by Stephen Wynkoop, Published: March 1997, and (7) Advanced PowerBuilder 6 Techniques by Ramesh Chandak.

Generally, the design methodology behind the present invention recognizes that the ability to create or innovate is, by itself, not sufficient to maintain a market advantage. Instead, a competitive advantage may be achieved by also recognizing that sorting and selecting important data from less relevant data enables a more coherent product strategy to be developed. That strategy is aided by the universal and timely availability of data, but within a structured environment. Corporate data should be added to a corporate data warehouse once, and all subsequent accesses are made

therethrough. Those accesses, however, must be customizable to allow the data to be output in a number of formats as required by recipients of that data.

Turning now to Figure 2, the inter-relationships of several of the processes of the present invention are generally illustrated as belonging to one of four different levels. As would be understood by one of ordinary skill in the art, those levels are conceptual and intended to facilitate discussion of those processes. Those processes, however, can be moved between levels or implemented to cross plural levels. In fact, in one embodiment, all the processes are implemented by a single executable program, with either a local user interface or a remote user interface (e.g., a Web interface). Moreover, the methodology described herein may be performed in a distributed fashion as well, with processes (or portions thereof) for creating information in a first location, tracking the information in a second location, and modifying it in a third. Thus, a computer system for implementing the present invention may also be implemented as a series of loosely or tightly coupled programs such that various portions of the system may be updated/upgraded without the need to make universal upgrades/changes. One such embodiment includes at least one application server provider.

Illustrated as part of Level 1 is the building of an Information Plan (e.g., a Product Optimization Plan (POP) that identifies information that has to be generated about a chemical, product, or combination). In the process of building a plan, the system receives the definitions of the goals to be achieved, the evidence needed to support that the goals have been met, and the issues involved in meeting those goals. Those definitions may be received either (1) on-line directly from various people or departments or (2) manually. Those definitions, however, must be such coordinated for review that (1) feedback can occur in building the plan and (2) the plan can be modified dynamically (using feedback as discussed in greater detail below).

After a plan has been created initially, information deliverables are defined, scheduled, and tracked in order to implement the goals of the plan. As part of designing and achieving those information deliverables, often the needs, evidence, and issues previously identified in Level 1 will have to be supplemented, revised, or discarded. To achieve this, the present invention utilizes a feedback loop between Level 1 and Level 2 such that the information plan reflects the information deliverables

and the information deliverables reflect the information plan. Part of that process also includes identifying and resolving issues that were previously unforeseen or not adequately addressed. The resolution of those issues, therefore, can result in supplementing, modifying, or discarding parts of the definitions of the plan. As
5 discussed above, a change in those definitions results in feedback to the understanding of how the plan itself is built.

In order to create at least some of the information deliverables of Level 2, modules that represent those deliverables must be authored, as shown in Level 3. Modules include, but are not limited to, drug activity results and regulatory
10 information. Those modules are then collected, reviewed, approved, and maintained. Changes in the modules are fed back to the definition, scheduling, and tracking functions of Level 2. This enables (1) schedules to be adjusted or reset and (2) deliverables to be tracked (e.g., the module was delivered at a specified time and did or did not meet its design objectives). As described above, if changes/adjustments are
15 required, feedback loops propagate those changes upwards in the hierarchy – potentially all the way up to Level 1.

Having built a collection of modules in Level 3, those modules (or a subset thereof) are used to assemble and produce documents supporting the goal of the plan. Generally, modules (or a subset thereof) are retrieved from digital storage (e.g., a file
20 server or a database) and preferably are reused in subsequent documents, either for the same Plan or different one. As with other levels, the assembly of reports (or dossiers in the pharmaceutical field) may uncover deficiencies in at least one previous level. Accordingly, as a part of the assembly and production phase, information may feedback to higher level(s) in the hierarchy.

25 Although paper-based reports are possible, the creation of reports according to the present invention need not be in paper form at all. In one embodiment of the present invention, paperless reports are generated and submitted (e.g., to a regulatory agency or a supervisor for review) electronically. This greatly reduces the amount of paper consumed and the publication and delivery time and costs associated with the use
30 of paper.

Approved
for MS
Submission

In general, the above architecture provides for flexible data sharing across a broad range of users. In order to achieve increased ease of use, it is advantageous, although not necessary, to provide a consistent user interface (e.g., a Web browser interface). Data sharing enables product information for a plan in all its phases (and, in fact, the plan itself) to be shared across an entire company (or multiple companies), if desired. As would be understood, the program implementing the user interface connects to a remote information server (e.g., a Web server, Gopher server or FTP server) and requests formatted and/or tagged data that is displayed locally.

As a separate aspect, security controls can be added to each of the processes of each of the levels. This facilitates limiting a first set of users to reading or viewing data while a second set of users is allowed to update information. Similarly, a third set of users can have the right(s) to authorize that various changes be made available for viewing or publishing, and a fourth set of users can be denied access altogether. As would be appreciated by one of ordinary skill in the art from this disclosure, any of various techniques can be used to authenticate users and restrict access to parts of the operation of the present invention and/or corresponding files. Secure socket connections and login scripts promote enhanced security, but other possible security measures can also be used. Additionally, in light of the desire to track document, module and template changes, authentication is used to track changes including who made the change and when. A complete history of each portion can then be examined upon request.

Depending on the number of people assigned to the various processes of the present invention, various degrees of independent operation can be achieved. While it is possible that a single person is responsible for each of the processes of Figure 2, the preferred embodiment uses plural people to increase concurrent activity. In light of the number of independent processes that can be performed on each of the different levels, the methodology of the present invention preferably utilizes at least one team of people for each level, where each team works independently of each other where possible. Complete independence would be advantageous; however, in light of the number of feedbacks within the system (and to provide continuity between levels), it is advantageous for at least one member of a team to belong to plural teams, one on each

Security

of two adjacent levels. This enhances the understanding of the problems or issues requiring feedback to a higher level. It also helps to articulate hidden assumptions that may have been made during the design and implementation phases that can help to speed up use at a lower level.

5 Turning now to the implementation of how information (e.g., a POP) is designed and tracked, a preferred embodiment utilizes a graphical user interface (GUI) (e.g., a Web browser interface) to act as the system front end. Using a Web interface reduces the design time required as compared to building a custom GUI, but a custom interface can be built in environments in which existing Web interfaces do not provide
10 all necessary features.

Using the GUI, needs (e.g., regulatory, commercial and manufacturing needs) are identified. An exemplary commercial need includes being able to state that product X is superior to product Y in advertising. Evidence to support such a statement may include comparative trials of products X and Y using one or more protocols. An
15 exemplary manufacturing need includes determining a specific supplier for a key intermediate used in the creation of a drug. The evidence needed to make the identification includes an equivalence study and the drug master file letter. Generally, though, the information plan is controlled by the needs of the "customer" using the plan at each level. (That "customer" may, however, change between levels and is often
20 within the same company at higher numbered levels.) At one level the customer is internal (e.g., the members of the project team or those interested in the products potential with the company but perhaps based in several countries). However, the exemplary goals are largely driven by the perceived needs of those who influence the decision to buy or use a product (e.g., in a medical community it is often doctors,
25 patients, payers and policy makers). Similarly, the evidence to prove those needs may also be driven by those deciding to purchase or use a product (e.g., in the medical marketplace the evidence is largely driven by customers and the various regulators that control distribution of medicines to the marketplace).

Using scripts (e.g., Active Server Pages) or middleware (e.g., ColdFusion), the
30 GUI presented to a user can be generated dynamically to include a level of detail commensurate with the level of skill and/or level of security clearance of the user.

Also, the interface can be updated periodically to reflect changes to any of the underlying data. This can be achieved by resubmitting a query to a database that stores each of the needs/goals, evidence, and issues that have been created or modified for the corresponding plan. That data can be filtered using any criteria established for the corresponding user (e.g., filtered only to show approved drafts or needs versus un-reviewed submissions).

Figure 3 shows a captured user interface illustrating that a product, XXX, has various sections that can be analyzed separately. For each of the sections illustrated in Figure 4, there is a corresponding database entry that tracks the status, version and date created for that section. As would be appreciated by one of ordinary skill in the art from this disclosure, any data repository can be tailored for use in the present invention. Although the data is illustrated as being stored in a database in Fig. 4, the information can be stored in a data file on a local computer or file server as well. In an embodiment using a data file, the data file can be in any one of many formats (e.g., comma delimited), but a self-describing and easily parsable format (e.g., HyperText Markup Language (HTML) or Extensible Markup language (XML)) is preferred. Moreover, any of the other data repositories described herein may be either (1) merged into the repository storing the data described above or (2) kept separate from that data (e.g., to increase redundancy, availability, etc.).

Similar to the tracking of sections, the present invention also tracks needs, evidence, and activities. As illustrated in Figure 3, the same user interface that was used to display sections and their statuses can also be used to display needs, evidence, and activities.

One need can require several activities and many bits of evidence. For example, a need might be 'proven efficacy in treating medical condition X'. By regulation, this proof is achieved through the results of at least two well-controlled clinical studies (in otherwords, two independent sources of evidence). Similarly, a need in a "once daily" medicine is to show that the body can respond to that limited frequency. The evidence required to support that need could include: (1) pharmacokinetic information about the amount of the drug in the body over 24 hours, (2) how much was absorbed versus how much was excreted, (3) pharamacodynamic

information that describes the duration of the desired pharmacologic effect, and (4) additional information that determines if the drug should be taken in the evening or daytime and with or without food.

5 Similar to the database structure illustrated in Figure 4, Figure 5 is a screenshot illustrating an exemplary database structure used to track the needs, evidence, and activities of a plan. By using the same identifier in both of the tables of Figures 4 and 5, the statuses of the sections can be correlated with the evidence for the same product.

10 As additional issues are identified, those issues can be addressed and resolved by using the feedback paths shown in Figure 2. Issues can be stored and retrieved from a log of issues (i.e., an issue log) for review at a later time. According to the present invention, the entries in the databases of Figures 4 and 5, can be updated, deleted, or created to reflect the newly resolved issues. For example, a section may need to be updated (and its version number changed) or an additional need may be identified. In either case, the dynamic generation of the interface of Figure 3 enables the product
15 information to be displayed to remote users as soon as the database is updated. (As noted above, for traceability reasons, earlier versions are not actually discarded from the database. Instead, the database contains entries for both versions. This enables any document to be regenerated from its constituent components or collections at any time.)

20 By using a computer system to track the various goals and/or modules related to a product or service, a company is enabled to provide earlier and more comprehensive input about a product's development, marketing and production. By using the present invention in the context of a drug management system, the needs of both domestic and international regulatory submissions can be addressed in parallel. Moreover, by using a web interface to remotely stored data, the data may be distributed among many remote
25 sites while appearing to be centrally located. That is, the storage location of the data components is transparent to the user of the Web interface.

Turning now to Level 2 of Figure 2, once the planning process for a new product is in place, information deliverables must be defined, scheduled and tracked (e.g., based on the information stored as part of Figures 3 through 5). Within Level 2,
30 two sub-levels are included that define how the functions of Level 2 are to be performed. As shown in Figure 6, the sub-levels are generally divided into (1) those

tasks that are to be tracked by the owner of the information deliverable and (2) those tasks that are the responsibility of a line manager. Generally, those tasks in the first group start with the identification of a series of outputs that correspond to the POP as presently defined (since feedbacks may change the design over the course of

5 development). Generally, the owner of the information deliverable should be able to easily identify what information he/she needs to provide to support the objectives of a project. Figure 6 illustrates five exemplary outputs from Level 2, although additional outputs are possible.

The process of the present invention schedules due dates corresponding to each
10 of the identified goals of the system; however, the due dates of the goals may include due dates for sub-goals that make up goals. Accordingly, changes in a sub-goal due date may, cause changes in a due date of a related sub-goal or corresponding goal. In establishing a regulatory submission goal, a study report can be created and tracked as a sub-goal.

15 One reason drug development can be complex is that many of the information elements needed for decision making and registration are affected by the content of other information elements. In order to facilitate module tracking, one embodiment of the present invention provides a visual representation (textual, graphic or a combination) of goal dependencies and/or goal/sub-goal dependencies so that changes
20 in planning can proceed smoothly as the information about the drug product matures. Examples of goals are a European Marketing Authorization Application (MAA) and a US New Drug Application (NDA). Examples of Information Deliverables which are components of these outputs are: Clinical Study Reports (several), Reproductive Toxicity Reports (several). The former would be a write-up of a clinical study and
25 would describe such aspects as: type of study (e.g., randomized double-blind multi-center), results obtained, and conclusions. The latter would describe a non-clinical study on a particular type of animal. Thus, the two applications would be assigned due dates as would all of the rest of the goals/sub-goals.

As seen in the Figure 6, the scheduling of the due dates is only a portion of the
30 tasks performed by an output owner for any information deliverable or output. The second portion of the tasks is tracking information deliverables and outputs. This

- tracking can either be a passive tracking (e.g., maintaining a list of statuses) or can be an active tracking (e.g., sending e-mails to responsible people or teams notifying them that they should have reached a particular milestone by a certain point in order to meet a projected due date). For example, if an information deliverable requires that a NMR test be run on a particular compound to authenticate its composition, the present invention can determine if there are sufficient slots open for the test to be run before the due date occurs. Similarly, if a due date is to occur in three days on a one man project, but there remains four man days of work, the system can notify a manager that additional people resources are needed to meet the goal or that overtime is required.
- 10 Both examples show how an active tracking system can help prevent additional delays which can "snowball" into delays across an entire project.

- As shown in Figure 6, the line manager is responsible for making sure the established due dates are met. The same overall design methodology is also visible in the authoring process illustrated in Figure 7. In order to implement many of the information deliverables defined by an output owner, the line manager may find it necessary to sub-divide a task further. As a result, a manager may choose to specify that a series of deliverables is to be implemented as a series of modules or even as a collection of modules. The production of the modules and/or collection is to be scheduled by the line manager and produced by programmers, engineers, scientists or marketing agents that perform data collection. The production of a deliverable may also require that additional details are provided about the task to be performed. This requires feedback to the output owner that is in charge of the corresponding output.
- 15
20

- As the information deliverables are produced, their production is tracked as discussed above, either passively or actively. However, if delivery due dates are missed, the system feeds back that information to enable dependent due dates to be updated as well. This overall tracking process facilitates generating the right deliverable (e.g., a dossier) at the right time, even in the face of changing deadlines. This process also facilitates early identification of deliverables that can be re-used by identifying what information is available and where it is located.
- 25

- Two possible user interfaces for defining and tracking deliverables are Gantt charts and Pert charts that are implemented as Active X or Java components embedded
- 30

within a web browser. An alternate embodiment uses a textual representation of the due date and the status (e.g., on-time, overdue, to be started, or completed). In a more active tracking implementation, a percentage of completion is measured to enable a line manager or output owner to more closely track whether corresponding due dates will be met.

Turning now to Level 3 of Figure 2, the processes of authoring individual modules, forming collections and maintaining modules are performed after information deliverables have been defined (or in response to changes in those deliverables). To facilitate the creation of modules, and in light of the fact that regulations change frequently, a (virtual) central repository of templates and guidance for using the templates is maintained. As was discussed above, the repository may either be actually central (i.e., the data is stored in a single location), or the repository may be virtual with the files being distributed under the control of a server that causes them to appear to be central. By storing them in the central repository, the templates and guidance are immediately delivered to a large number of users, thereby relieving the system administrator of the task of searching out all the old copies of the templates and guidance.

To gain access to the repository, a user logs into the repository, using either a dedicated repository tool or a web interface. Such a log-in process can either be transparent (e.g., using a single-sign on function or a locally stored cookie) or manual (e.g., requiring that the user input a user name and password). As shown in Figure 8A, during a user's first log-in, the system automatically allows a user to customize the user's access to the repository (e.g., by specifying an identifying name and default location and business area). Subsequently, upon user request, the user can use an interface (e.g., as shown in Figure 8B) to update the user's previously stored preferences.

As shown in Figure 9, a user can request, by selecting the "Template List" hyperlink, a filtered list of the templates stored in the repository that are available for download by the authenticated user, filtered by at least one criteria (e.g., by business area of the user or personal preferences). Figure 10A illustrates a partial list of templates that are available to the user. In the leftmost columns, status indicator 220A

and type indicator 220B, each implemented as either a graphic or text, identify the status and type of template that is available. Figure 10A shows the names 230 of exemplary templates (i.e., the portions of reports that start out empty and are filled in prior to being submitted). In order to save the user from manually finding the file
5 corresponding to the name 230, the name 230 preferably also acts as a hyperlink to the file directly. Generally, there is a virtual central repository that provides access to the latest copy to ensure that the user uses the most up-to-date copy available. If that copy is unavailable via the hyperlink, the registered owner of the copy is notified of the unavailability. One embodiment of this notification process includes automatic logging
10 of the unavailability and auto-sending. In another embodiment, the user experiencing the error is given a chance to specify a message to the owner.

Figure 10B provides a legend for the status indicators 220A used in Figure 10A. Illustrated statuses include: (1) Pending, (2) New, (3) Updated, (4) Information, (5) Current, (6) Withdrawn, and (7) Deleted. However, other statuses are also possible. In
15 fact, a database stores the status types such that new status codes can be defined dynamically (i.e., without code recompilation).

Figure 10C provides a legend for the file types of the corresponding templates. One of the most commonly used templates is a Word template. In an exemplary Word implementation, a module template based on the normal template encapsulates the
20 default formatting options for Word documents. This ensures that the resulting document meets a specified standard for content, quality and presentation. Moreover, by using toolkits within a Word environment, a user may quickly and correctly add symbols and other formatting to documents or components. This facilitates migration between different versions of the environment and even paper sizes (e.g., A4 to letter).

25 Generally, however, template and modules are defined by reuse, not necessarily by how the information is presented. For example, other modules include information stored in SAS databases and listings and other templates include Excel spreadsheets. Similarly, many templates created in Word have embedded in them tables and text that are derived from other format types.

30 Generally, the present invention employs a building block approach to assembling and viewing scientific data and information that can then be reused in many

ways (e.g., constructing regulatory and corporate documents, supporting business decisions, sharing learning across projects).

5 A module is one building block. Each module is discrete information that exists as a file or files within an electronic repository or shared electronic storage area in a form that can be individually identified, viewed, and retrieved. A module is released for use by a specific accountable individual who has either authored the module content in a standardized format or obtained output from a validated computer or imaging system.

10 Modules are classified by their type. Module types define consistent and comprehensive content that is flexible enough to be applied to different kinds of output (regulatory and corporate documents). For example, the synopsis of a clinical study report is a module type. There is an agreed template and guide for many module types that instruct the author on what content is expected and how it should be displayed. Once the template is populated with the required information it becomes a module that is then released into an electronic repository. If a particular project generated 10 clinical studies, then there would be 10 clinical study report summary modules all of the same module type and authored using the same template and guide.

20 The size and scope of modules vary according to their content and their use in producing information outputs. Good design of module types and their associated templates and guides is the key to eliminating unnecessary duplication of information in the repository. Moreover, by providing a central repository that is transparently accessible, duplication is also avoided.

25 A module type defines information that is required as a component of a regulatory or corporate output or is needed in business decision making. Information defined by each module type share one or more of the following characteristics: (1) it is reused in multiple outputs, (2) it is valuable within an organization outside of where it was produced (e.g., timely access to this information is needed to make business decisions or to author other modules), (3) it is a complete file from a computer system that must be kept discrete for validation purposes (e.g., SAS tables and output from Clinical statistical systems), (4) it is defined by a template that ensures different authors provide consistent information, (5) it meets a specific output need that cannot be met by

another module containing the same information (e.g., if the information needs of European and US regulators are so different that it is easier to define a module type specific for that output).

Not everything is a module, however. A module type should not be created if
5 limited by any of the following conditions: (1) the overhead in defining the module type, its template and/or guide and maintaining the associated modules in the electronic repository clearly exceeds the value of module access and reuse, (2) information does not impact business decisions outside of its functional area, (3) the same information is defined elsewhere in the electronic repository in a useable form (e.g., already defined
10 by another module type), and (4) in the pharmaceutical environment, the information is not needed for regulatory or corporate documents.

Examples of module types include, but are not limited to, major groups such as: CMC Module types, Non-clinical Module types, Biotechnology Module types, Clinical Study Report Module Types, and Regulatory Administration and Labeling Module
15 Types. Examples of CMC Module Types include: Description of Synthesis, Flow Chart of Synthesis, Quality Control During Synthesis, Container/Closure System, GMP Certification, Physical and Chemical Characteristics, Elucidation of Chemical Structure, Impurities in the Drug Substance, Drug Product Specification – Release, Drug Product Specification - Shelf life, Dissolution – Method, Representative Batch
20 Formula, Manufacturing Process, Method of Manufacture & Packaging, Manufacturing Flow Diagram, In-Process Controls, Clinical Trial Formula, Batch Analysis for Drug Substance, Batch Analysis for Drug Product, Stability Studies Report Drug Substance, Ongoing Stability Studies – Product, Control Tests on Intermediate Products, and CMC Summary.

25 Changes to the layout and content of templates and guides may occur in response to new regulatory or customer needs, changes in corporate document standards or an agreed change in accepted practice by the relevant scientific division. Templates and guides will be version controlled and the history of change recorded. However, the most readily accessible template and guide will be the most recent
30 version.

Authors create modules by populating module-type templates with information and data in accordance with agreed guides. Once the author populates the template with the required information, a new module of information is born. Generally, a module can be defined conceptually by the equation below.

5

Module = Template for specific Module Type + Information & Data

The author is responsible for releasing the module into the shared repository after seeking whatever local business approval is required. After release, the module is
10 available for viewing and use by others outside of the business area where it was generated.

Modules are time-based in that new versions may be created as new information becomes available or existing information is changed or updated during drug development. Some types of modules are dynamic with frequent updates and some are
15 static in that information does not change after its creation. Examples of dynamic modules are: (1) product or compound stability where data is updated at designated intervals (e.g. 6 months, 1 year, 2 years), (2) continuous updates of batch record information to a summary module, and (3) information for regulatory reports submitted at prescribed intervals (e.g. IND Annual Report, post-approval safety summaries).

20 Usually, only the author incorporates changes or updates to a module. Once a change has been approved, the updated information is released to the repository as the next numbered version of the module. Older versions may be kept 'live' in the system or 'retired' to archival storage depending on business needs.

Attributes are assigned to each module during creation and at the time of release
25 into an electronic repository. When taken together, these attributes uniquely identify the information contained in a module. Users access modules of interest either by searching on module attributes or by following logical information paths and hierarchies.

Modules of information are associated together into collections for two
30 purposes:

1) Approval together for use because one or more modules cannot be released for use in isolation. For example, the summary module of the collection is a module because of its reuse in many outputs, but it cannot be released for use until the information is reconciled with the other modules of this collection. Moreover, some modules may not ever be published as part of a study report but still need to be consistent with information contained in the other modules of the collection.

2) Assembled together in preparation for publishing. Modules are assembled into collections for publishing in one of the following ways: (1) simple ordering of existing modules, (2) insertion of an entire module into the body of another module where the desired content, order and format cannot be achieved by simple concatenation (e.g., different modules are pasted into the middle several different locations of a new document), (3) assimilated in part into another module. This implies a 'cut and paste' activity but with strict control over what information is selected and who can author in this manner.

Although assimilation of information from an existing module into another results in some duplication of information within a repository or electronic storage areas, it is sometimes a necessary pre-publishing step because of the limitations of current publishing tools. Exemplary regulatory dossiers may include a chemistry, manufacturing and controls sub-section.

Module collections are assembled by a coordinating author, reviewed by all contributing module authors and released to the repository as an approved collection for wider use (e.g. published report, viewing summary document packages, incorporation into regulatory submissions). One of the attributes of a module collection is the list of modules and their versions that form this collection. Changes to the collection are tracked by version number in the same way changes to modules are managed.

Templates, however, can further be augmented with descriptive information that identifies what the information is, not just how it is represented. In one embodiment, templates are implemented using a descriptive grammar (e.g., a Data Type Definition (DTD) for an Extensible Markup Language (XML)) that specifies the relationship

between entities in the final module. For example, a DTD can define that a module must have certain attributes to be valid. Subsequently, formatting can be done intelligently using at least one style sheet.

Returning to Figure 10A, if guidance is available for utilizing the retrieved
5 templates, indicators 220C describe what type of guidance is available (e.g., using the legend of Figure 10B). Although not shown in Figure 10B, exemplary video guidance formats include, but are not limited to, MPEG, AVI, and Shockwave formats. To facilitate easy access, a hyperlink is provided for each form of guidance that is available. By providing guidance, an end-user spends less time on figuring out the
10 formatting and structure of the information deliverable and is freed to concentrate on how to generate the underlying technical information. (In one embodiment of the guidance, the guidance and the template are integrated into a single document to facilitate review of the guidance.)

As shown in Figure 10A, indicators 220D identify to a user that some templates
15 also have more detailed information describing the corresponding template. Examples of stored details include, but are not limited to, Name, Status, File Extension, Source path, Business Area, Functional Area, Subgroup, Ownership, Authoring Country, Module type, Doc Ref, and Comments. Some of those fields are searchable as is described in greater detail below. Exemplary interfaces showing details of a document
20 template and guidance, respectively, are illustrated in Figures 10D and 10E. (The template may be referenced in multiple business areas, but the template need only be stored in a single location per repository. The reference need only point to the real file location.) A corresponding interface also may be provided for displaying to which business area a template belongs. That interface includes a feedback hyperlink to
25 enable comments on the templates or guidance to be submitted. Feedback can be submitted in the form of comments using a Web interface or using an e-mail and associated attachments. In at least one embodiment, feedback is linked to the owner of the template or guidance to enable direct feedback, comments or questions. This creates greater accountability for those templates or guidance modules and enhances
30 the management of the requested changes.

As the number of templates grows, it becomes increasingly difficult to select the appropriate template or guidance for a particular task. As a result, as shown in Figure 9, a "Search" hyperlink is provided in one embodiment to facilitate selecting a subset of the templates and/or guidance stored in the repository. Upon clicking on the "Search" hyperlink, a search interface, such as the interface shown in Figure 11, is displayed to the user. The user fills out at least one form field in the interface and selects the "Search" button. The repository then finds all the templates (or guidance) that meet all of the specified conditions. Other examples of fields that can be used in a search are: product name, date, author, and team/group/department. In a more flexible search interface, the user may also specify whether the search is a boolean AND or a boolean OR of the search terms. The user may also clear the form fields in their entirety by selecting the "Clear" button.

Just as the number of templates may be burdensome in general, so may it be burdensome to remember the search terms that produced a desired result. Accordingly, in one embodiment of the present invention, a list of search results terms or partial search terms can be stored as a filter for later retrieval. For example, Figure 12 illustrates the results of the search of Figure 11. Figure 12 also includes a button labeled "Save Search as Personal Filter." In response to selecting that button, the user is requested to specify a name to identify the search terms (acting as a filter) for later retrieval. By saving the filter rather than the result, the search is re-run later, and the most up-to-date results are retrieved. When a user wishes to retrieve a saved filter, the user can select a "Stored filters" button (not shown) and specify (e.g., using a list or text box) a stored filter. The user is then presented with the results of the retrieved query that is re-run.

In an alternate embodiment, the user has the option of saving the results (i.e., the list of templates or guidance produced by a filter) instead of the filter itself. This enables a user to recall the same set of templates or guidance as was previously obtained, even if a characteristic of the search (e.g., the template status) changes in between the two searches.

Common searches can also be stored as filters. Such searches can be specified by a system administrator upon review of the most frequently run searches. Exemplary

filters include searches stored by business group, document type, or a combination of parameters.

Just as a user may wish to add filters to manage templates, a user may wish to delete stored filters as well. As is shown in Figure 13, a user can select the saved filters
5 that are to be removed from the user's profile.

The central repository also provides a method for tracking the number of times that a given template or guide is used. This enables developers to focus on templates that are used frequently. Those templates are assigned higher priority during updating and/or upgrading than other templates since those templates are likely to be re-used
10 again quickly. This tracking can be used for quality control as well since it ensures that users are selecting the most recent templates as frequently as they should be. To further aid in the efficient use of the repository, the interface of Figure 9 also provides a "Site Feedback" hyperlink and a "Help" hyperlink. Selecting those hyperlinks provides a user with feedback and help interfaces such as are shown in Figures 14 and 15,
15 respectively.

Although the above discussion has concentrated on the use of templates by users, such templates are additionally managed by an administrator (using a tool with an interface as shown in Figure 16A). The legend for the icons of the tool is provided in Figure 16B. Using such an interface, a new template can be added (as shown in
20 Figure 17) and searches for existing templates can be executed (as shown in Figure 18). As a result of a search, a user can perform at least two operations (as shown in Figure 19). Either (1) a new version of an existing templates can be created, or (2) meta-data of existing templates can be edited (as shown in Figure 20). When creating a new version of an existing template, the administration tool (1) automatically updates the
25 version number by 1 to track changes to a template, (2) assigns a new document ID, and (3) records the date and time of the saved transaction. An Administrator can withdraw or delete a template at any time after saving by selecting the template and changing its status to withdrawn or deleted (as shown in Figure 21). Once a deleted or withdrawn transaction is saved, the corresponding document is no longer available to
30 end-users.

A back-end processor handles the insertion of new templates into the virtual central repository. In one embodiment, the template to be added is processed prior to insertion (e.g., the template is scanned for viruses, checked for compatibility with other components (e.g., a current word processor version) and placed in a global replication area that enables the template to be copied to a global application server). During a replication phase, in addition to starting the replication process, the results of the replication are checked to ensure that the central server represents the current state of the templates/guidance. If the replication worked correctly, a snapshot of the available templates is taken such that if database access is unavailable (e.g., an back-end Oracle database fails to provide access to the template characteristics), then the user can still find the location of the most up-to-date copy of any template/guidance using its name.

As an additional administration function, the interface of Figure 16A allows the selection of a Report Manager that produces a report interface (e.g., the interface shown in Figure 22). Such an interface can be used to view the results of replication operations for distributing templates to various locations. Such an interface can also track its own operations so that it can generate a transaction report.

As a portion of the information management of the present invention, reuse of information between reports should be facilitated. For example, if information is managed properly, then modules to be used in a report for a UK regulatory agency can be re-used when generating a US regulatory document. Accordingly, reusable templates are defined as a starting point for generating the modules that form reports. After the reusable templates are retrieved from a virtual central repository (e.g., a file server or a database) as described above, the templates are filled in with product-specific information (thus becoming a module) and assembled into a document.

The assembly process may require the modification of reusable modules. Although manual (i.e., by hand) distribution of module corrections is possible, another embodiment utilizes an e-mail-based review process. In that review process, an authorizing or reviewing agent receives an e-mail containing (1) an indication of what document is to be authorized or reviewed and (2) a link to that document. In one embodiment of the present invention, Adobe Exchange is used to view the document and add annotations thereto.

The control of the status of modules supports the review and approval process. For example, an unreviewed module starts out with a draft status, and the same module is eventually (1) withdrawn or (2) promoted to an approved or a released module after all outstanding issues have been resolved (e.g., data and formatting are correct). At
5 each of those stages, the new modules (or simply new or newly approved versions of old modules) are added to a central module repository as well. Such a central module repository may be either on the same machine or a different machine as the template repository. In one embodiment of the present invention, to reduce search time and complexity, the modules are stored in a hierarchical storage system with the hierarchy
10 being illustrated as a series of folders and documents. Access to the individual folders and modules is achieved using either native file system access control support (e.g., as supplied on a Unix or NT workstation) or using a separate security component (e.g., using the socket connections and login scripts described above).

Document components can be combined to form new modules, collections or
15 documents (e.g., in light of differences in international regulations). (Module templates and collection templates may facilitate the combination process.) Module templates have no real lifecycle but are managed by the system. Modules have a lifecycle (i.e., starts either blank or from a module template as draft) and are the main content of the system. Collection templates embody a collection of modules. They have no real
20 lifecycle but are managed by the system. Module collections are collections of modules that have lifecycles. That is, if a draft collection is temporary it can be deleted. However, if a released collection is retained it cannot be deleted and modules composing the collection are recorded.

Combinations of document components may require that information be
25 reformatted so that it is suitable as a printed publication (as opposed to changing the contents of the information). This process is known as "pre-publishing" and includes multiple activities. Additional "white space" that occurs between grouped modules is removed to provide a continuous flow between the corresponding sections. Similarly, since components have headings, tables, figures, footnotes, endnotes, headers and
30 footers that are individually numbered, those elements must be renumbered to maintain continuity between grouped components. Once that consolidation process has been

completed, a table of contents referencing any subset of those elements can be generated as well. The result is a pre-published collection that can be incorporated into a final document.

In one embodiment of the present invention, the pre-publishing process removes
5 any module-specific information. In such an implementation, the pre-publishing process must be re-run after changes to any of the individual components have been made so that the pre-published collection remains synchronized with the corresponding component. This approach is different that editing the pre-processed report and having to determine how the changes in the pre-processed report correlate back to individual
10 components that formed portions of the report.

Stripping out information, however, does not imply a loss of accountability. Even after pre-publishing, it is preferred to be able to identify within a pre-published document each of the components used to generate the pre-published document. To do so, unique identifiers and version numbers may be utilized.

15 As described above, to maintain consistency between components and collections, the system maintains revision histories and trees identifying (1) which collections were generated with which specific versions of which components, and (2) which components are used to generate which collection generally. This enables any version of any document to be reproduced by re-using the parameters that generated the
20 original.

Just as with searching for templates, searching for collections may become increasingly difficult as the number of collections increases. As a result, filtering and searching techniques, as described above, can also be applied to the management of collections. In one embodiment, collections are searchable using criteria including, but
25 not limited to, a name and a unique identifier.

In addition, in other embodiments, the computer interface of the present invention provides other functions, either individually or in combination. Examples of such features include: (1) a menuing system for importing modules or module collections, (2) a status updater for updating the status of a module or collection, (3) a
30 withdraw option for removing a module or collection from the central repository, (4) an unlock option for allowing changes to a previously released module or collection

(thereby changing the status as well), (5) a registration function, (6) a check status option, (7) bind/unbind options for components, (8) an option to create an AdHoc collection or a collection template, and (9) show options. The show options can include any one or a combination of options to show: (1) component versions, (2) components
5 with contents later than a particular version, (3) collections in which a component is used, (4) bound components, and (5) unbound components.

The second to last logical component of Level 4 assembles and publishes reports (or dossiers in a pharmaceutical environment). That publication can either be in paper or electronic form. Generally, the publishing function coordinates the merger of
10 various data sources (e.g., pre-published collections, images, tables, templates, and non-pre-published collections). Each of the data sources can be represented by a different file format (e.g., Word, Excel, TIFF, ASCII, Postscript, and PDF formats), depending on the characteristics of the data source. In light of the possible distributed nature of the file repository, the merger of data sources may necessitate that a source
15 file be obtained from a remote site. Such a file may be obtained using any file transfer mechanism (e.g., using a file system client, HTTP, or FTP). As with pre-published components, the merger requires that the coordination of the components provide a consistent numbering between headers, footers, etc. and generate a final table of contents. Depending on the complexity of the report being generated, each component
20 may have to be analyzed several times (i.e., the report may require several passes) to properly calculate the final pagination of a document.

In one embodiment of that logical component, CoreDossier is used to generate the output in any format specified by the user that is supported by CoreDossier. Examples of output formats include, but are not limited to, Postscript, PDF, HTML,
25 and XML, although XML outputs require a style sheet to produce a final output.

The last logical component of Level 4 of Figure 2 is an interface for browsing and viewing previously output reports (or dossiers). As described above, the individual components of a document are tracked. Accordingly, one feature of the report browser is the ability to request a list of component files that were used to make a specified
30 report. Such a list may include the status of each of those component files.

As discussed above, for large report generation requests, the report generation process can be rather lengthy. Accordingly, the report browser also provides a mechanism for requesting the status of a print request (e.g., a determination of what percentage of the document has been finalized and what sections remain to be
5 finalized).

In light of the sensitivity of the data in the previously generated reports, the security features discussed above, or other security features, are used to provide access control to those reports. One possible location for their storage is a Documentum system. In one embodiment, special security clearance is required to browse through
10 parts of the submission within the system as a method of regulatory review (a replacement for reviewing a paper document). In that embodiment, navigation would be facilitated for the reviewer, and the reviewer would know when they have the latest data. In fact, if appropriate the reviewer can look at underlying data that supports the information being reviewed. The sponsor could monitor review of the dossier, and the
15 system provides a common place (e.g., message board or instant messaging) for quick exchange of questions and answers about the data that could be readily documented.

Figure 23A illustrates an exemplary set of needs that could be tracked according to the present invention. By specifying the needs as at least (1) a need type, (2) a need output type, and (3) a date by which the need must be met, the present invention
20 enhances the automation of the need tracking. One way to specify such needs is to add standardized (structured) Property flags such that a user can identify the Outputs (if any) in which the Need must be addressed (covered, supported). An alternate structure for specifying the needs is as XML-based objects. Once a series of needs have been specified, they can then be tracked as a series of outputs, as shown in Figure 23B.

25 Using this system, Activities are assigned (linked) to Information Deliverables. Moreover, in a preferred embodiment, reports are printed that tabulate Needs-Activities-InfoDeliverables-Outputs. That report can then be used by the project team to manually verify that the Outputs are indeed capturing the right Activities (and evidence) to support the IP Need. This report would be especially helpful where the
30 links are complex (e.g., non-study Activities such as special statistical analyses or

discussions of extrapolations; or CMC Activities linked to large/complex Information Deliverables).

Although the above-description has, at times, been directed to the manufacture of drug-based products, the present invention is directed to all products generally (e.g.,
5 medical devices that are not controlled by a regulatory agency, medical devices that are controlled or need approval, and biological compounds and/or agents)

Obviously, numerous modifications and variations of the present invention are possible in light of the above teachings. It is therefore to be understood that, within the scope of the appended claims, the invention may be practiced otherwise than as
10 specifically described herein.

CLAIMS:

1. A computer program product, comprising:
a computer storage medium and a computer program code mechanism
embedded in the computer storage medium for causing a computer to track information
5 regarding a plan, the computer program code mechanism comprising:
a first computer code device configured to select a plan requiring a report to be
generated;
a second computer code device configured to select a report type for the plan;
a third computer code device configured to track completion of information
10 goals required to generate the report;
a fourth computer code device configured to select document components of the
report from a repository of versioned document components; and
a fifth computer code device configured to submit a report generation request to
an information server to generate the report of the report type using the selected
15 components.
2. The computer code device as claimed in claim 1, wherein at least one of the
first through fourth computer code devices is implemented using a Web browser.
3. The computer code device as claimed in claim 1, wherein at least one of the
first through fourth computer code devices is implemented using a plug-in for a Web
20 browser.
4. The computer code device as claimed in claim 1, wherein at least one of the
first through fourth computer code devices is implemented using an interface created
dynamically using middleware.
5. The computer code device as claimed in claim 1, wherein the fourth
25 computer code device includes a filter for reducing a number of selections.
6. The computer code device as claimed in claim 1, wherein the fourth
computer code device includes a number of selections filtered at at least one of the
information server and the repository.
7. The computer code device as claimed in claim 1, further comprising a sixth
30 computer code device configured to generate a copy of the report and store the copy in
the repository.

8. The computer code device as claimed in claim 7, further comprising a seventh computer code device configured to store in the repository a list of the components used to generate the report.

9. The computer code device as claimed in claim 1, wherein the fifth computer code device comprises a sixth computer code device configured to retrieve the selected components from plural distributed computers.

10. The computer code device as claimed in claim 1, further comprising a sixth computer code device configured to produce at least one of the document components from a centrally stored template.

11. The computer code device as claimed in claim 10, further comprising a seventh computer code device configured to provide textual help on using the template.

12. The computer code device as claimed in claim 10, further comprising a seventh computer code device configured to provide audio-based help on using the template.

13. The computer code device as claimed in claim 10, further comprising a seventh computer code device configured to provide video-based help on using the template.

14. The computer code device as claimed in claim 1, wherein the fourth computer code device comprises a sixth computer code device configured to select a pre-published component as one of the selected components.

15. The computer code device as claimed in claim 1, wherein the repository comprises a database.

16. A computer system for tracking information regarding a plan, the system comprising:

means for selecting a plan requiring a report to be generated;

means for selecting a report type for the plan;

means for tracking completion of information goals required to generate the report;

means for selecting document components of the report from a repository of versioned document components; and

means for submitting a report generation request to an information server to generate the report of the report type using the selected components.

17. A computer-implemented method comprising:

- 5 (a) selecting a plan requiring a report to be generated;
(b) selecting a report type for the plan;
(c) tracking completion of information goals required to generate the report;
(d) selecting document components of the report from a repository of versioned document components; and

(e) submitting a report generation request to an information server to generate
10 the report of the report type using the selected components.

18. The method as claimed in claim 17, wherein at least one step of steps (a) through (d) is implemented using a Web browser.

19. The method as claimed in claim 17, wherein at least one step of steps (a) through (d) is implemented using a plug-in for a Web browser.

15 20. The method as claimed in claim 17, wherein at least one step of steps (a) through (d) is implemented using an interface created dynamically using middleware.

21. The method as claimed in claim 17, wherein step (d) of selecting comprises filtering to reduce a number of selections.

22. The method as claimed in claim 17, wherein step (d) of selecting comprises
20 filtering a number of selections at at least one of the information server and the repository.

23. The method as claimed in claim 17, further comprising generating a copy of the report and storing the copy in the repository.

24. The method as claimed in claim 23, further comprising storing in the
25 repository a list of the components used to generate the report.

25. The method as claimed in claim 17, wherein step (d) of selecting comprises retrieving the selected components from plural distributed computers.

26. The method as claimed in claim 17, further comprising producing at least one of the document components from a centrally stored template.

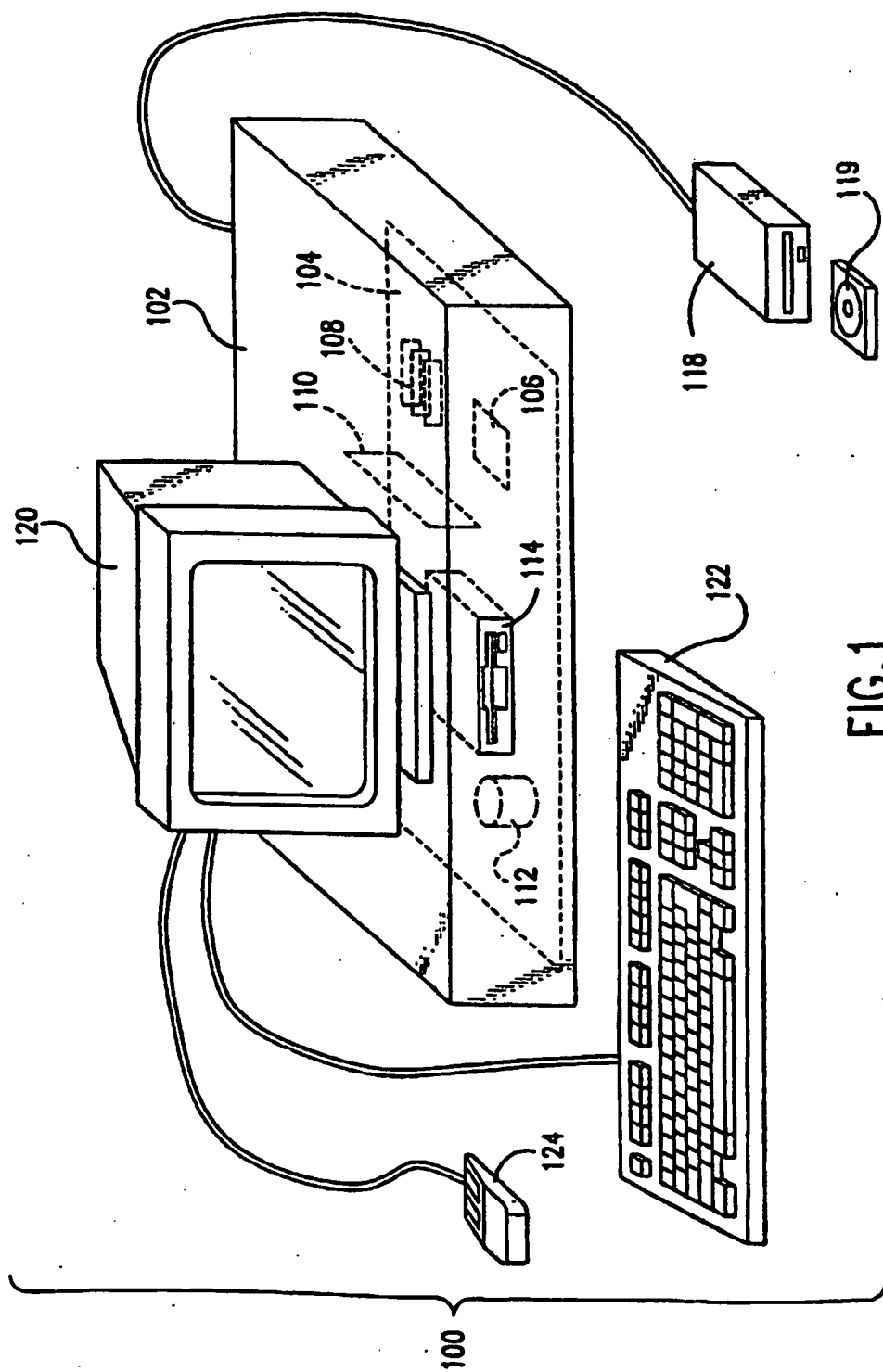
30 27. The method as claimed in claim 26, further comprising providing textual help on using the template.

28. The method as claimed in claim 26, further comprising providing audio-based help on using the template.

29. The method as claimed in claim 26, further comprising providing video-based help on using the template.

5 30. The method as claimed in claim 26, further comprising selecting a pre-published component as one of the selected components.

31. The method as claimed in claim 17, wherein the repository comprises a database.



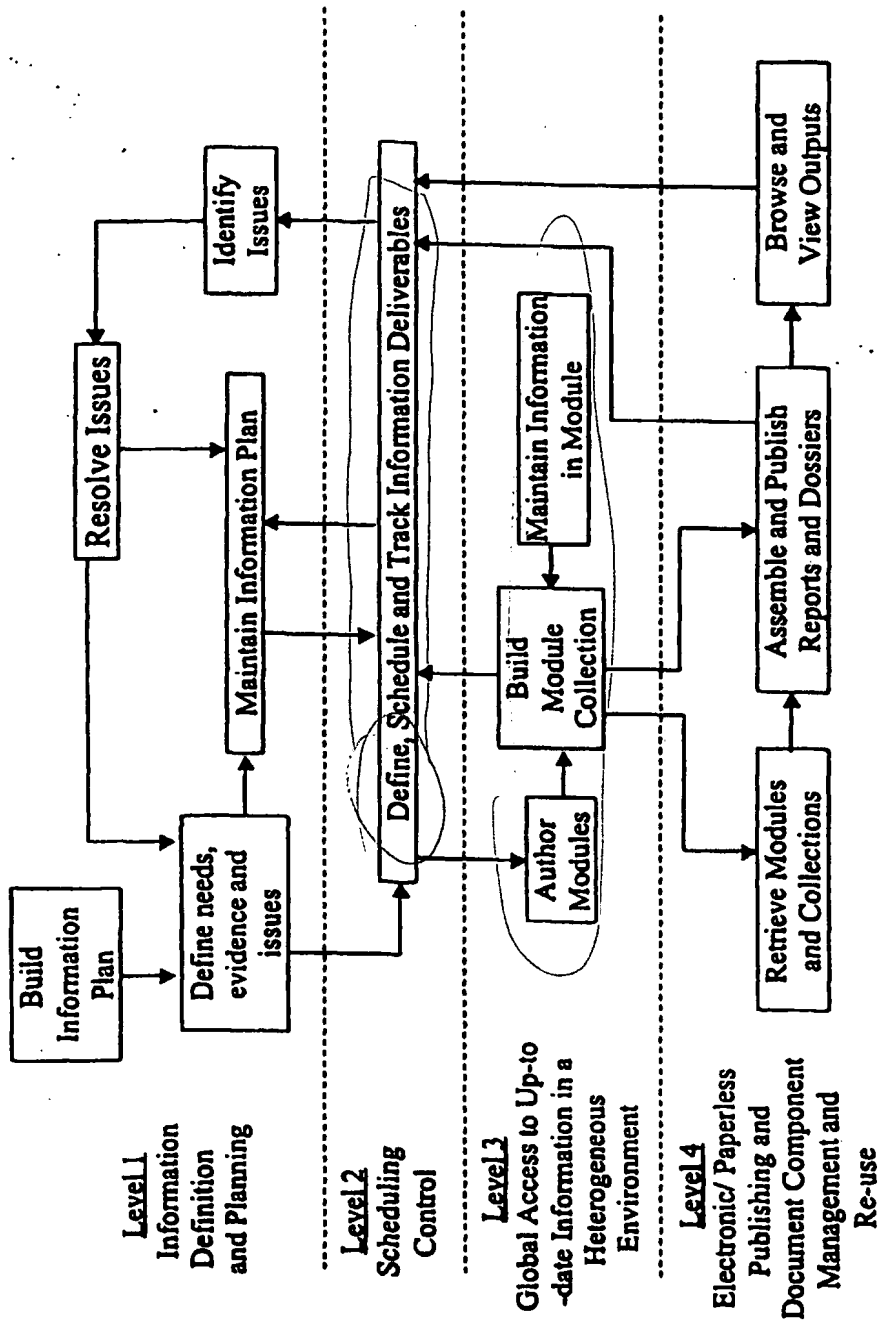


Figure 2

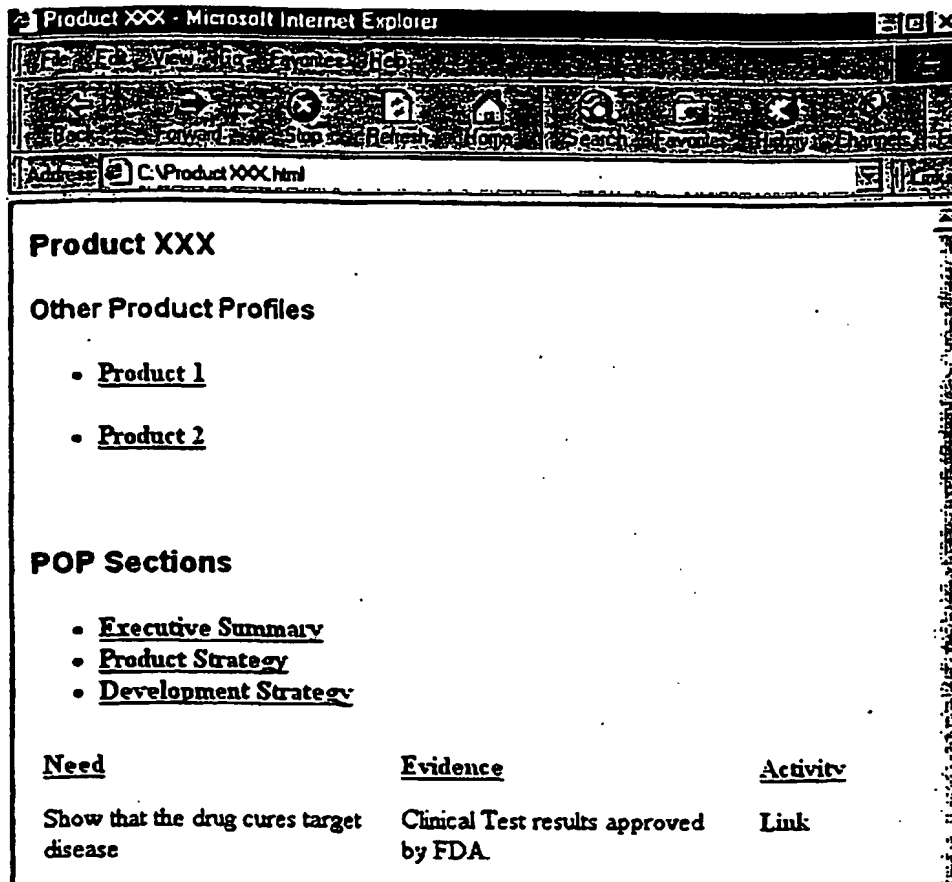


Fig. 3

Microsoft Access

File Edit View Format Window Help Record View

Section: Table

ID	Development Strateg	Not reviewed	Ver	Ver Date
123	Development Strateg	Not reviewed	0	4/1/00
123	Executive Summary	Draft	0.01	4/1/00
123	Product Strategy	Draft	0.02	4/1/00

Unique ID for selecting a record

Fig. 4

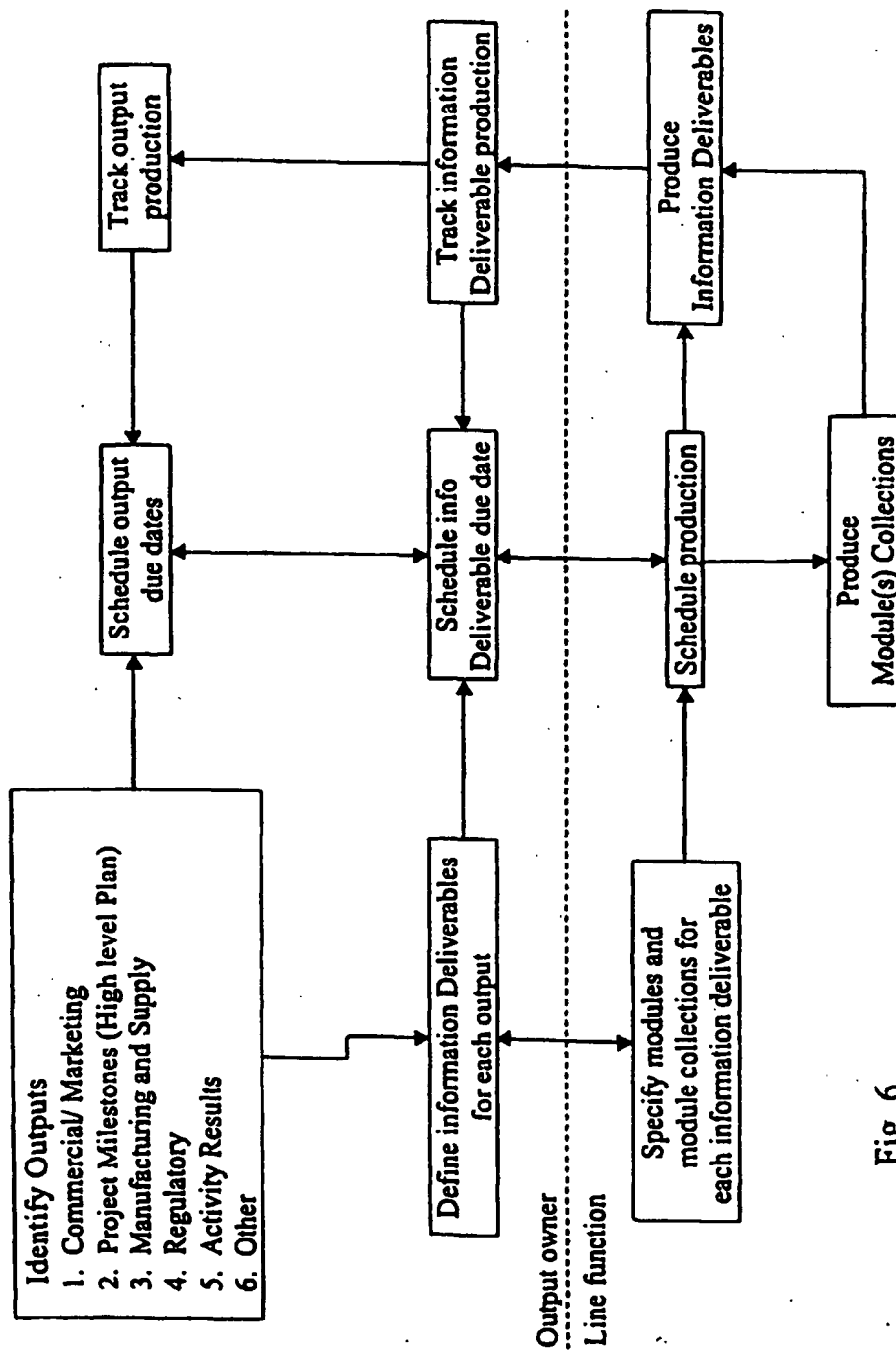


Fig. 6

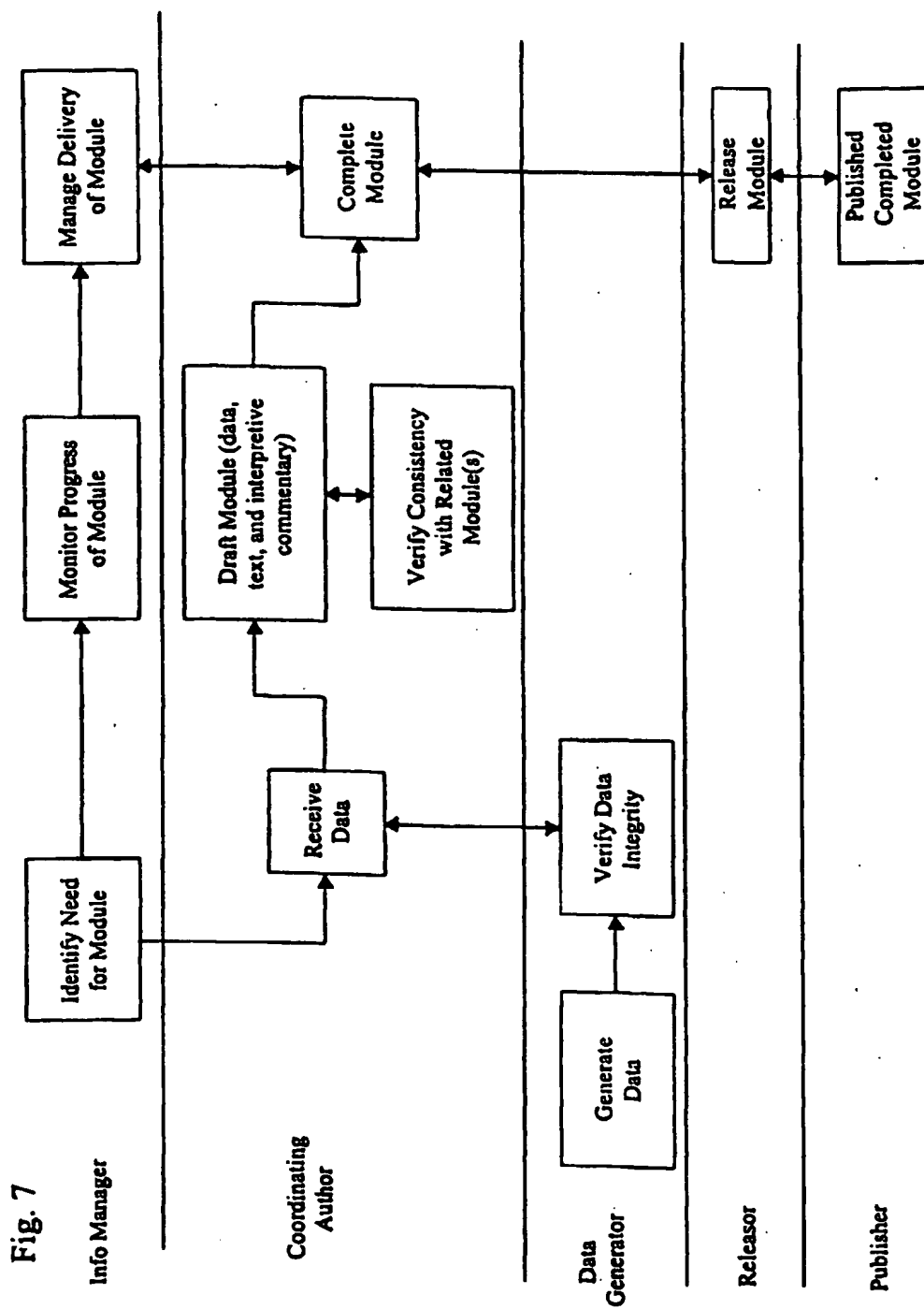


Figure 8A

You appear to be using TAG for the first time.

Please enter your name and select your preferences.

Your User ID	ABC1234
Your First Name	Robert
Your Last Name	Martens
Home Location	US-RTP-1 <input type="checkbox"/>
Current Location	US-RTP-1 <input type="checkbox"/>
Business Area	Non-Clinical <input type="checkbox"/>

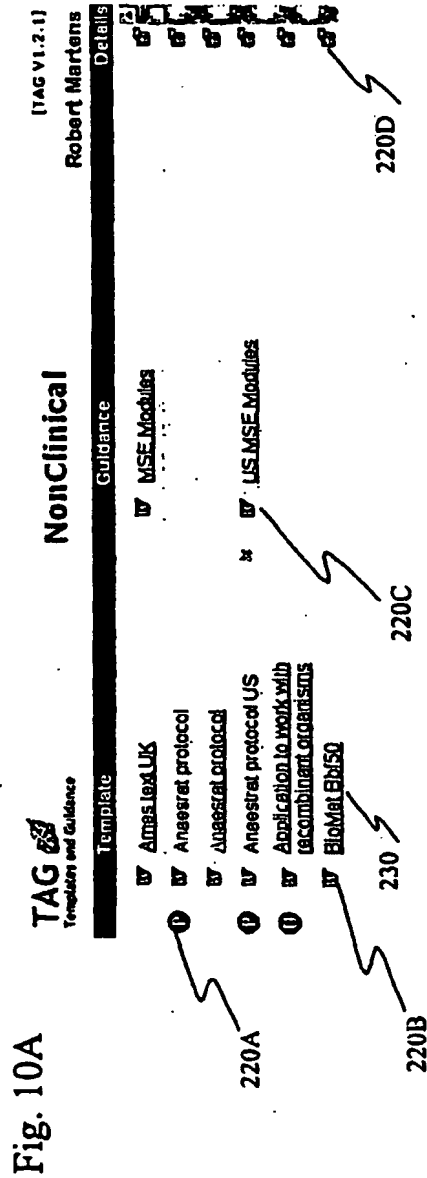
Login

Figure 8B

Update Your Preferences	
Your User ID	ABC1234
Your First Name	Robert
Your Last Name	Martens
Home Location	US-RTP-1 <input checked="" type="checkbox"/>
Current Location	UK-Stevenage - 2 <input checked="" type="checkbox"/>
Business Area	UK-Stevenage - 3
	UK-Stevenage - 4
	UK-Stevenage - 5
	UK-Stockley Park
	UK-Ware - 1
	UK-Ware - 2
	UK-Ware - 3
	UK-Ware - 4
	US-RTP-1 <input checked="" type="checkbox"/>
	US-RTP-2 <input checked="" type="checkbox"/>

[Template List](#) | [Filter](#) | [Search](#) | [Options](#) | [Site Feedback](#) | [Help](#)
 The information provided within this site is intended for internal use only.
 GW Intranet Copyright ©1998 Glaxo Wellcome Inc. All rights reserved.

Fig. 9








Status	Description	Icon
Pending	Entered into the system but not yet available for viewing. The name of the file will appear, but the link will not be active.	
New	A file that is newly available for viewing.	
Updated	Changes have been made to an existing file.	
Information	Allows the TAG System Administrator(s) to enter information about the file (ex: File under review, check back for updates.)	
Current	The is available in the system. It has been available longer than the "new" or "updated" period.	None
Withdrawn	The name of the file will appear, but the link will no longer be active. The file is scheduled to be removed from the server.	
Deleted	The file is not available in the list of files to view. The file has been deleted from the server.	N/A

Figure 10B











	Microsoft Word Document
	Microsoft Excel Document
	Microsoft PowerPoint File
	Web Site (Using Internet Explorer)
	Web Site (Using Netscape)
	Web Document (Using IE)
	Web Document (Using Netscape)
	Acrobat PDF File
	Micrografx FlowCharter
	Guidance Internal to the Template

Figure 10C

Template

④ Clinical Expert Report
Business Area Reg Admin / Main Lib, Clinical
Status Updated Document
Date Last Updated 7/27/00 8:05:51 AM
Version 1
File Type Word Document
Comments Template updated to change text size to point 11 and to update the distribution lists.

Functional Area	Sub Group	Country	Document Type	Output
Clinical_clinical development	Clinical_None	Clinical_All	Clinical_Expert Report	Clinical_All
Clinical_clinical pharmacology				
Clinical_health outcomes				
Clinical_medical affairs				

 **Document Feedback**

Guidance

Guidance Documents (in Pathfinder)
Business Area Reg Admin / Main Lib, Clinical
Status Current Document
File Type Web Site



Guidance Feedback

Search Applies To: ☒ Template ☐ Guidance

Business Area

Display Name (contains)

File Type

Status

SubGroup

Functional Area

Authoring Country

Output Type

Template Module Type

Figure 11

Template

Guidance

Details

List of all Template files where Business Area = NonClinical AND File Type = Word Document AND Template Name text contains 'Ames Text Uk'

☒ Ames text UK☒ MSE Modules

Figure 12

Select the filters you wish to DELETE from your saved profile

Filter Name	Filter Description
Bob's Filter	Business Area = NonClinical AND File Type = Word Document AND Template Name text contains 'data flow'

Figure 13

From: Robert Martens (ABC1234) Home Location: US - RTP - 1 Current Location: US - RTP - 1 Business Area: NonClinical	
Name: TAG Feedback	
Comments:	<input type="text"/>
<input type="button" value="Send"/> <input type="button" value="Reset"/>	

Figure 14

Welcome to TAG on-line help

Here you will find information to help you navigate your way through the TAG system. We have tried to anticipate your questions about the system and have divided Help in to specific categories to facilitate you locating information quickly and easily. Choose a help section from the list below:

Help Categories	
What is TAG?	Details
Writing Submissions Documents	Status Flags
TAG Quick Reference	Options
Getting Started	Searching
Locating and Viewing Files	Filters
Browser Basics	Feedback
Icons	Special Functions

Figure 15

Template and Guidance Administration Tool

File Edit View

Viewing Mode

Template Manager Guidance Manager Report Manager

Find Template: Ames test UK V-2

Template Name: Ames test UK

Version: 2

Doc Ref:

Status: Current Document

File Ext: DOC

Source Path: \\A:\4545\SE\Departmental Areas\Business Strat

Guidance: MSE Modules V-2.1

Comments: April Review

Business Area: NonClinical

Owner(s): Goodchild, Andrea (ajg654C)

Country: UK

Output: NonCln_Report

Sub Group: NonCln_Genetic Tox

Functional Area: NonCln_MSE

Template Module Type: NonCln_Report Component

Template ID: 6851 Date Of Last Update: 07/04/2000 9:21:12 AM

Figure 16A







-  Enters a new document in TAG
-  Saves a Transaction
-  Edit a document
-  Search for existing document
-  Cancel the current transaction
-  Exit the TAGAT Tool

Figure 16B

Template and Guidance Administration Tool

Template Editing Mode

Template Manager Guidance Manager Report Manager

Template Name: Ames test UK

Version: 2

Doc Ref:

Status: Current Document

File Ext: DOC

Source: \\uk145\MSSE\Departmental Areas\Business Strat

Guidance: MSE Modules V-2.1

Comments: April Review

Business Area: NonClinical

Owner(s): Goodchild, Andrea (ajg554)

Country: UK

Output: NonClin_Report

Sub Group: NonClin_Genetic Tox

Functional Area: NonClin_MSE

Template Module Type: NonClin_Report Component

Template ID: 685 Date Of Last Update: 07/04/2000 9:21:12 AM

Figure 17

Template and Guidance Administration Tool

Viewing Mode

Template Manager Guidance Manager Report Manager

Find Template: Ames test UK V-2

Ames test UK V-2

Ames test UK

Anaesthet protocol V-1.1

Anaesthet protocol V-3

Anaesthet protocol

Anaesthet protocol V-2

Anaesthet protocol

Anaesthet protocol US

Current Document

File Ext: doc

Business Area: NonClinical

Owner(s): Goodchild, Andrea (ajg554)

Country: UK

Output: NonClin_Report

Sub Group: NonClin_Genetic Tox

Figure 18

Template and Guidance Administration Tool

File Edit Help

Template Editing Mode

Template Manager Guidance Manager Report Manager

Find Template: Ames test UK V-2 Business Area: NonClinical

Ownerlist: Goodchild, Andrea (alp554E)

Template Name: Ames Edit Options

Version: 2

Doc Ref:

Status: Current

File Ext: DOC

Source:

Do you want to create a New Version or Edit Existing Version?

New Version Edit Existing Cancel

NonClinical Genetic Tox

Figure 19

Template and Guidance Administration Tool

File Edit Help

Template Editing Mode

Template Manager Guidance Manager Report Manager

Business Area: NonClinical

Ownerlist: Goodchild, Andrea (alp554E)

Country: UK

Doc Ref:

Status: Current Document

File Ext: DOC

Path: \\ukz445\\MSE\\Departmental Areas\\Business Strat

Guidance: MSE Modules V-21

Comment: April Review

Template ID: 3251

Date Of Last Update: 07/04/2000 9:21:32 AM

NonClinical Genetic Tox

NonClinical MSE

NonClinical Report Component

Figure 20

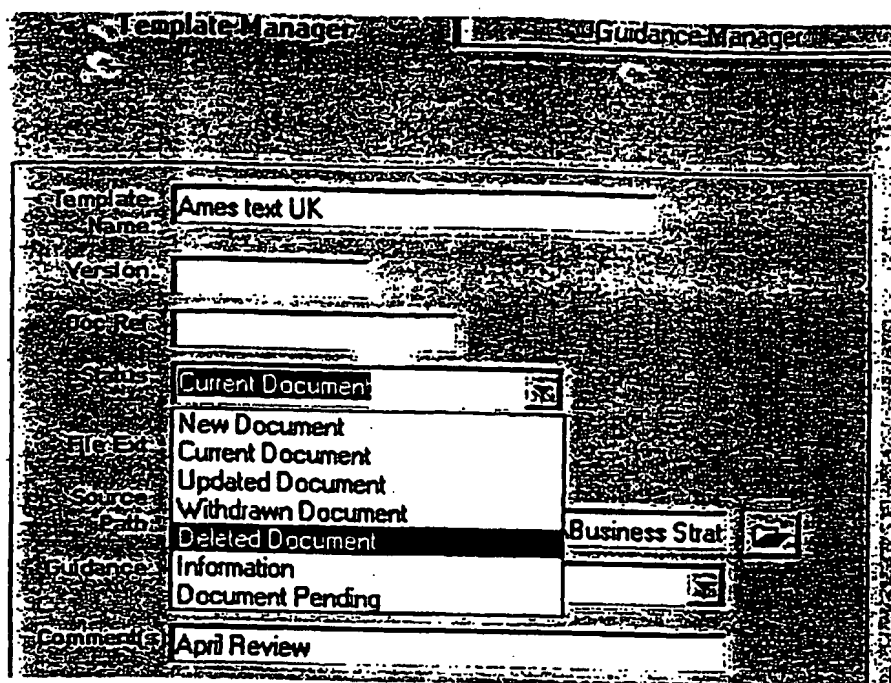


Figure 21

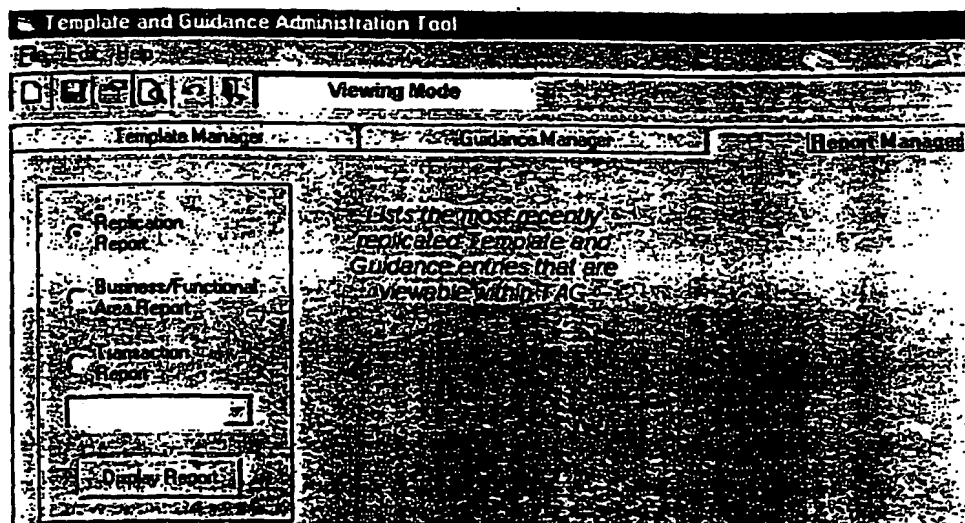


Figure 22

Fig. 23A

NEED1 is Claim A (Commercial Need)

NEED1 must be covered in

OUTPUT1 (MAA) by DATE (1Q02)

OUTPUT2 (NDA) by DATE (1Q02)

OUTPUT3 (Publication) by DATE (3Q02)

OUTPUT4 (abstract) by DATE 4Q01

OUTPUT5 (US Sales Training Manual) by DATE (3Q02)

OUTPUT6 (Submission to pricing authority) by DATE (2Q02)

NEED1 is supported by

ACT (Study X), and documented in INFODEL31

ACT (ISE analysis), and documented in INFODEL51

ACT (ISE extrapolation from other drugs in class), and documented in INFODEL52

NEED2 is Claim B (Commercial Need)

NEED2 must be covered in

OUTPUT1 (MAA) by DATE (1Q02)

OUTPUT6 (Submission to pricing authority) by DATE (2Q02)

NEED2 is supported by

ACT (Study Y), and documented in INFODEL32

ACT (Study Z), and documented in INFODEL33

NEED3 is that tablet coating must be made by new aqueous formulation xxx (Supply Need)

NEED3 must be covered in

OUTPUT1 (MAA) by DATE (1Q02)

OUTPUT2 (NDA) by DATE (1Q02)

OUTPUT7 (PAI Package) by DATE (1Q02)

NEED3 is supported by

ACT (Stability studies), and documented in INFODEL91

ACT (Bioequivalence study), and documented in INFODEL11

NEED4 is that hydroxy acid starting material must be supplied from Upjohn (Supply Need)

NEED4 must be covered in

OUTPUT1 (MAA) by DATE (1Q02)

OUTPUT2 (NDA) by DATE (1Q02)

NEED4 is supported by

ACT (Verify acceptability of DMF), and documented in INFODEL92

NEED5 is that data must be available to show best dose was selected (Regulatory Need)

NEED5 must be covered in

OUTPUT1 (MAA) by DATE (1Q02)

OUTPUT2 (NDA) by DATE (1Q02)

OUTPUT4 (US Sales Training Manual) by DATE (3Q02)

OUTPUT6 (Submission to pricing authorities) by DATE (2Q02)

NEED5 is supported by

ACT (Phase 1 PK study), and documented in DEL12

ACT (Phase 2 dose-range study), and documented in DEL21

Fig. 23B

OUTPUT1 (MAA) must be ready by 1Q02, and must include at least the following Information Deliverables in order to support NEEDS 1, 2, 3, 4, 5

INFODEL 31, 51, 52

INFODEL 32, 33

INFODEL 91, 11

INFODEL 92

INFODEL 12, 21

OUTPUT2 (NDA) must be ready by 1Q02, and must include at least the following Information Deliverables in order to support NEEDS 1, 3, 4, 5

INFODEL 31, 51, 52

INFODEL 91, 11

INFODEL 92

INFODEL 12, 21

OUTPUT3 (Publication preparation package) must be ready by 3Q02, and must include at least the following Information Deliverables in order to support NEED 1

INFODEL 31, 51, 52

OUTPUT4 (abstract) must be ready by 4Q01, and must include at least the following Information Deliverables in order to support NEED 1

INFODEL 31, 51, 52

OUTPUT5 (US Sales Training Manual) must be ready by 3Q02, and must include at least the following Information Deliverables in order to support NEEDS 1, 5

INFODEL 31, 51, 52

INFODEL 12, 21

OUTPUT6 (Submission to pricing authority) must be ready by 2Q02, and must include at least the following Information Deliverables in order to support NEEDS 1, 2, 5

INFODEL 31, 51, 52

INFODEL 32, 33

INFODEL 12, 21

OUTPUT7 (PAI package) must be ready by 1Q02, and must include at least the following Information Deliverables in order to support NEED 3

DEL 91, 11